Journal of
Marine Science and Engineering

MDPI

Article

Autonomous Underwater Pipe Damage Detection Positioning and Pipe Line Tracking Experiment with Unmanned Underwater Vehicle

Seda Karadeniz Kartal * and Recep Fatih Cantekin

Electrical and Electronic Engineering Department, Zonguldak Bulent Ecevit University, 67100 Zonguldak, Turkey; recep_cantekin@hotmail.com
* Correspondence: sedakaradeniz@gmail.com

Abstract: Underwater natural gas pipelines constitute critical infrastructure for energy transportation. Any damage or leakage in these pipelines poses serious security risks, directly threatening marine and lake ecosystems, and potentially causing operational issues and economic losses in the energy supply chain. However, current methods for detecting deterioration and regularly inspecting these submerged pipelines remain limited, as they rely heavily on divers, which is both costly and inefficient. Due to these challenges, the use of unmanned underwater vehicles (UUVs) becomes crucial in this field, offering a more effective and reliable solution for pipeline monitoring and maintenance. In this study, we conducted an underwater pipeline tracking and damage detection experiment using a remote-controlled unmanned underwater vehicle (UUV) with autonomous features. The primary objective of this research is to demonstrate that UUV systems provide a more cost-effective, efficient, and practical alternative to traditional, more expensive methods for inspecting submerged natural gas pipelines. The experimental method included vehicle (UUV) setup, pre-test calibration, pipeline tracking mechanism, 3D navigation control, damage detection, data processing, and analysis. During the tracking of the underwater pipeline, damages were identified, and their locations were determined. The navigation information of the underwater vehicle, including orientation in the x, y, and z axes (roll, pitch, yaw) from a gyroscope integrated with a magnetic compass, speed and position information in three axes from an accelerometer, and the distance to the water surface from a pressure sensor, was integrated into the vehicle. Pre-tests determined the necessary pulse width modulation values for the vehicle's thrusters, enabling autonomous operation by providing these values as input to the thruster motors. In this study, 3D movement was achieved by activating the vehicle's vertical thruster to maintain a specific depth and applying equal force to the right and left thrusters for forward movement, while differential force was used to induce deviation angles. In pool experiments, the unmanned underwater vehicle autonomously tracked the pipeline as intended, identifying damages on the pipeline using images captured by the vehicle's camera. The images for damage assessment were processed using a convolutional neural network (CNN) algorithm, a deep learning method. The position of the damage relative to the vehicle was estimated from the pixel dimensions of the identified damage. The location of the damage relative to its starting point was obtained by combining these two positional pieces of information from the vehicle's navigation system. The damages in the underwater pipeline were successfully detected using the CNN algorithm. The training accuracy and validation accuracy of the CNN algorithm in detecting underwater pipeline damages were 94.4% and 92.87%, respectively. The autonomous underwater vehicle also followed the designated underwater pipeline route with high precision. The experiments showed that the underwater vehicle followed the pipeline path with an error of 0.072 m on the x-axis and 0.037 m on the y-axis. Object recognition and the automation of the unmanned underwater vehicle were implemented in the Python environment.

Keywords: unmanned underwater vehicle; convolution neural network; underwater pipe line tracking; underwater pipe damage detection; navigation of unmanned underwater vehicle

check for updates

Citation: Kartal, S.K.; Cantekin, R.F. Autonomous Underwater Pipe Damage Detection Positioning and Pipe Line Tracking Experiment with Unmanned Underwater Vehicle. *J. Mar. Sci. Eng.* **2024**, *12*, 2002. https://doi.org/10.3390/jmse12112002

Academic Editor: Weicheng Cui

Received: 19 August 2024
Revised: 31 October 2024
Accepted: 31 October 2024
Published: 7 November 2024

1. Introduction

1.1. Motivation

Unmanned underwater observation vehicles are critically important for various military and civilian applications. These unmanned vehicles are used in civilian fields such as underwater mapping, port security, geological geophysics, and fisheries, and in military areas for mine detection, enemy ship detection, ship safety, coastal security, and human detection, as well as in underwater cable and pipeline laying operations [1–3]. The highly variable nature of the underwater environment makes these operations challenging. Using unmanned underwater vehicles instead of human divers in long-term operations in dark and deep waters is both safer and more cost-effective due to the potential risks to human life [4,5]. Divers can only remain submerged for a limited time during any underwater operation due to the risk of hypothermia from prolonged exposure. For this reason, the duration of underwater operations tends to be extended. To eliminate the negative aspects arising in similar underwater operations, unmanned underwater vehicles have started to be preferred for underwater tasks.

Underwater natural gas pipelines form a critical infrastructure for energy transportation. Any damage or leakage occurring in these pipelines can create serious security risks. Proper monitoring and damage detection facilitate the early identification and prevention of potential hazards. Additionally, underwater natural gas pipelines can pose a direct threat to marine and lake ecosystems. Any leakage or damage can lead to environmental pollution and harm aquatic life. Timely damage detection can help minimize environmental impacts. Damages to underwater natural gas pipelines can also cause interruptions in the transmission of natural gas, leading to serious operational problems in the energy supply chain. Continuous monitoring and damage detection are crucial for maintaining operational continuity. Furthermore, damages, repairs, and interruptions in the pipeline can lead to significant economic losses for energy companies. Timely damage detection can reduce costly emergency interventions and increase operational efficiency. There are legal and regulatory requirements for underwater natural gas pipelines in many systems. These requirements mandate regular monitoring of the pipelines and the performance of damage detection.

In this study, damages on an underwater pipeline were detected using an unmanned underwater vehicle. As important as detecting damages on the pipeline is knowing the locations of these damages. If the location of the damage is unknown after it has been detected, finding the locations of the damages would require additional time due to the extensive length of the pipelines. In this study, autonomous features were added to the unmanned underwater vehicle, enabling autonomous tracking of the damaged pipeline underwater, and while tracking the pipeline, the damages were diagnosed using artificial learning and their locations were determined. This study involves the autonomous tracking of an underwater pipeline of a certain length by an unmanned underwater vehicle (UUV), focusing on detecting and localizing damages on the pipeline during this tracking. The key focus areas and objectives of the study are as follows:

- Autonomous Pipeline Tracking: The remote-controlled UUV autonomously follows the underwater pipeline using its navigation system.
- Damage Detection: Damages on the pipeline are identified using images captured by the UUV's camera, processed through a Convolutional Neural Network (CNN) algorithm.
- Damage Localization: The location of the detected damage is determined by correlating it with the vehicle's INS (Inertial Navigastion System) based navigation data, providing an accurate position.
- Real-World Application: The system is tested and validated for the usability of underwater pipeline maintenance and monitoring in real-world environments.

1.2. Related Works

For successful execution of unmanned underwater operations, both a well-defined underwater environment and accurate knowledge of the underwater vehicle's position are necessary [6–8]. Therefore, recognizing underwater objects and navigation of underwater vehicles are important. Various types of identification algorithms are available to identify an object. In this study, the convolutional neural network (CNN) training algorithm, a deep learning method, was used. CNNs are used in various fields including image classification, object tracking, object recognition, exposure estimation, text detection and recognition, visual projection detection, action recognition, scene labeling, speech processing, and natural language processing [9]. Other training models require a large amount of prior knowledge at the end of training to achieve high accuracy in object recognition. However, in the CNN model, input data is provided to the model without the need for feature extraction or creation processes. Since the CNN model trains by altering the depth and width of the input image, it determines the features of the image and makes accurate assumptions [9,10]. CNN training requires significant computational resources, but several methods have been developed to address this issue [11–27]. The most important of these developments is the development of a CNN model trained using the ImageNet dataset in 2012. This model produced more accurate image classifications than previous methods [9]. He and Zhang in 2018 suggested predicting movements from an image with CNN [28]. Pertusa and Gallego in 2018 used CNN for common object identification on smartphones [29]. In 2015, Li and Shang used the fast region-based CNN (R-CNN) algorithm for underwater fish detection [30]. In 2017, Gomez Chavez and Mueller predicted body posture using the Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) method [31]. The CNN algorithm continues to be used in classification studies today [32–34]. Another current machine learning algorithm is the Support Vector Machine (SVM). The SVM algorithm emerged in 1995 [35]. It is a high-performance algorithm frequently chosen for regression and prediction problems [36]. To date, SVM has been used in various contexts including battery life prediction, housing price forecasting, and predicting potential inflation [37–42]. Although SVM is more commonly chosen for classification problems, Smola and others have shown it can also be used for regression problems [43]. The algorithm used for regression problems is named Support Vector Regression (SVR). SVR has been applied in various regression problems such as motion prediction [44], electric load forecasting [45], and enhancing the performance of filters [46–48] . In this study, CNN and SVM were used in a hybrid manner. CNN served as a feature extractor and SVM as a classifier to identify the target object.

Various methods are employed in the literature for detecting damage to underwater pipelines. While image processing and deep learning algorithms (CNN) can detect damages on pipelines with high accuracy, methods such as sonar and acoustic imaging offer the advantage of scanning larger areas. Techniques like magnetic flux leakage (MFL) and ultrasonic testing (UT) detect corrosion and cracks within the internal structure of the pipes, while fiber optic sensors and vibration-based monitoring methods provide continuous monitoring and early detection capabilities. However, each method has its advantages and limitations depending on environmental conditions, pipeline material, and the size of the damage. While modern image processing techniques such as CNN analyze visual data, acoustic and magnetic methods like sonar and MFL can scan broader areas regardless of environmental conditions. The choice of method depends on the characteristics of the pipeline and environmental factors. Unmanned underwater vehicles (UUVs) are effective tools for detecting underwater pipeline damage when integrated with various detection technologies. For visual-based detection, deep learning algorithms such as CNN can offer higher accuracy compared to traditional methods like sonar and magnetic techniques. The methods used for underwater pipeline detection and the advantages and limitations of these methods are presented in the comparison table in Table 1. As seen from the comparison, the use of both the unmanned underwater vehicle and the CNN algorithm in this study has significantly enhanced performance. In addition, to facilitate fault detection in subsea pipelines, transient test-based techniques were used to generate safe and small overpressure waves, allowing the detection of anomalies such as leaks and wall deterioration [39,40].

Table 1. Performance comparision of pipeline damage detection methods.

Method	Key Strengths	Key Limitations	Paper Citation
CNNs (Optical Image)	High accuracy with image data Automatic Feature xtraction Handles compex surface defects Adaptable and scalable with large datasets	Require large training datasets Computationally intensive	[14–16]
Acoustic Imaging	Effective in deep or turbid water Covers large areas	Lower resolution, noisy data Difficult to detect small defects like cracks or corrosion Complex data interpretation and prone to noise	[17]
Magnetic Flux Leakage	Good for internal defects and corrosion Real time monitoring	Only works for metallic pipelines Ability to detect external surface damage like cracks or dents Limited compared to CNN-based visual inspection	[18]
Ultrasonic Testing	High precision for internal defect, non-invasive	Requires physical contact or close proximity to the pipeline surface, which can be difficult in certain underwater conditions. Doesn't provide as detailed information on surface defects as CNNs do with optical imagery	[19]
Fiber Optic Sensing	Real-time, continuous monitoring, detects strain and deformation	Installation is complex and expensive, as it requires physically embedding the sensors along the pipeline. It doesn't provide detailed visual feedback, unlike CNN-based methods.	[20]
Vibration based Monitoring	Sensitive to structural changes, covers long distances	Influenced by external factors such as sea currents, environmental noise, or nearby operations, making data interpretation challenging. Unlike CNNs, it doesn't provide a direct visual assessment of the damage.	[21]
Visual Inspection Using Underwater Vehicle	Real-time inspection High-resolution data Can reach inaccessible areas Lower cost Continuous monitoring Long-term stability	High cost Limited battery life	[22–26]

Another important factor for the successful execution of unmanned underwater operations, such as underwater pipeline tracking and underwater pipeline damage detection, is the localization of the underwater vehicle [49]. Due to the attenuation/damping of electromagnetic waves underwater, high-accuracy global positioning systems cannot be used. With inertial measurement systems, linear and angular position information is derived from the measured acceleration and angular velocity of the underwater vehicle [50]. Some studies use integrated navigation systems for high accuracy and continuous data transmission. If an INS-GPS integration system is to be used, a surface platform synchronized with the underwater vehicle is essential [51]. In this study, the navigation information of the underwater vehicle was obtained from integrated gyroscopes, magnetic compasses, accelerometers, and pressure sensors.

Despite extensive work on aerial and terrestrial object tracking, there is much less research on underwater object tracking. This is due to the various challenges of working underwater and the degradation of underwater visual data quality, which varies depending on light refraction, water depth, color, and nature. In 2013, Min Li and colleagues presented a method for underwater object identification and tracking based on multi-beam sonar imaging [52]. In 2016, Filip Mandic and colleagues combined sonar and USBL (Ultra Short Baseline) measurements to develop an autonomous surface vehicle and perform underwater object tracking [53]. They developed a filter that combines USBL and sonar image measurements to obtain reliable object tracking predictions even when sonar or USBL measurements are unavailable or erroneous. In addition to object tracking, they focused on adapting only the desired region within the sonar image using the tracking filter's covariance transformation to improve object identification and filter out erroneous sonar measurements. In 2016, Xianbo Xiang and colleagues proposed a method using magnetic sensing to autonomously track underwater buried cables with a three degrees of freedom (3-DOF) autonomous underwater vehicle [54]. They used feedback linearization technique to design a simplified cable tracking controller based on the geometric relationship between the vehicle and the cable by creating a specialized magnetic line of sight guide. In 2020, Caterina Bigoni and Jan S. Hesthaven suggested a simulation-based decision strategy with machine learning techniques for anomaly detection and damage localization [55]. In 2021, Kakani Katija and colleagues proposed using an underwater vehicle for the visual tracking of deep-sea animals controlled by machine learning [56]. In their study, they presented an integrated tracking algorithm using machine learning that includes multi-class detectors and 3D stereo imaging to track underwater animals over extended periods. There are many studies like these focused on underwater object tracking, and the research continues. Studies such as continuous autonomous tracking and imaging of great white sharks with an autonomous underwater vehicle, and performance analysis of existing underwater object tracking algorithms and dataset creation are available [57,58].

1.3. Contribution

In this study, an autonomous system was developed to monitor underwater pipelines using an unmanned underwater vehicle (UUV). Damages to underwater pipelines were detected with high accuracy through the CNN algorithm. The precise location of the detected damage was determined using the UUV's navigation data, ensuring precise positioning. The system's usability for the maintenance and monitoring of underwater pipelines was tested and validated. There is no experimental study in the literature that autonomously tracks underwater pipelines with a UUV while simultaneously detecting pipeline damage using deep learning algorithms and pinpointing the damage's location. The research gap, contribution, and innovation of this study can be summarized in the following key points:

- Underwater Damage Detection with Deep Learning: Autonomous systems for detecting damages in underwater pipelines are limited in the literature, and the use of deep learning algorithms such as CNN in this area is rare. This study aims to automatically detect damages in underwater pipelines using deep learning algorithms, offering a faster and more accurate approach compared to manual methods.
- Autonomous Damage Detection and Navigation Relationship: The ability to correlate the damage location with the vehicle's INS-based navigation data is an innovation in terms of damage detection and accurate positioning of underwater vehicles. There is no example in the literature where these two capabilities are combined in real-time and autonomously.
- Experimental Application: Another strength of this study is the experimental application of the CNN algorithm. Unlike simulation or theoretical studies, this experimental part, which demonstrates real-world applicability, distinguishes this research from others in the literature.
- Potential for Industrial Application: This technology could be used for the monitoring and maintenance of underwater infrastructures, such as oil, gas, and other underwater pipelines, with the potential to improve the management of these infrastructures.

In conclusion, this study aims to address the gap in the literature by offering an innovative approach to real-time and reliable underwater pipeline damage detection using autonomous underwater vehicles and deep learning algorithms.

1.4. Organization

The paper is organized as follows. The unmanned underwater vehicle used in our underwater experiments is introduced in Section 2. The method used for the underwater pipe damage detection experiment, involving a convolution neural network, and the experimental results of the damage diagnosis are explained in Section 3. The experiment on underwater autonomous pipe tracking and damage location detection is detailed in Section 4. In Section 4, the unmanned underwater vehicle navigation, autopilot, underwater damage location detection, and experimental results are also presented sequentially. In final, the paper is concluded in Section 5.

2. Unmanned Underwater Vehicle Used in Experiments

In this study, the remotely operated underwater vehicle (ROV) that will be endowed with autonomous features consists of a user computer, an operator console, and cable section. The unmanned underwater vehicle is equipped with two forward thrusters, one on the left and one on the right, a vertical thruster, and a camera that can rotate 180 degrees. The forward thrusters provide forward movement and yaw orientation, while the vertical thruster provides diving movement. The operator console is used for controlling the vehicle and transferring the data obtained from the vehicle to the computer. The cable ensures data and power transmission between the underwater vehicle and the operator console. In this study, the experimental equipment used in the pool experiment is presented in Figure 1 [59].

The unmanned underwater vehicle used in this study measures 42 cm in length, 33 cm in width, and 27 cm in height, with a total mass of 10 kg. The vehicle is designed to operate at depths of up to 200 m. Its moments of inertia are 0.223 kg/m^2 along the x-axis, 0.225 kg/m^2 along the y-axis, and 0.06 kg/m^2 along the z-axis. The maximum surge speed of the vehicle is 1.5 m per second. The vehicle's sensor suite includes an Inertial Navigation System (INS) from the MPU-6000 series, featuring accelerometers, gyroscopes, and a depth sensor (MS5837-30BA), as well as a camera.

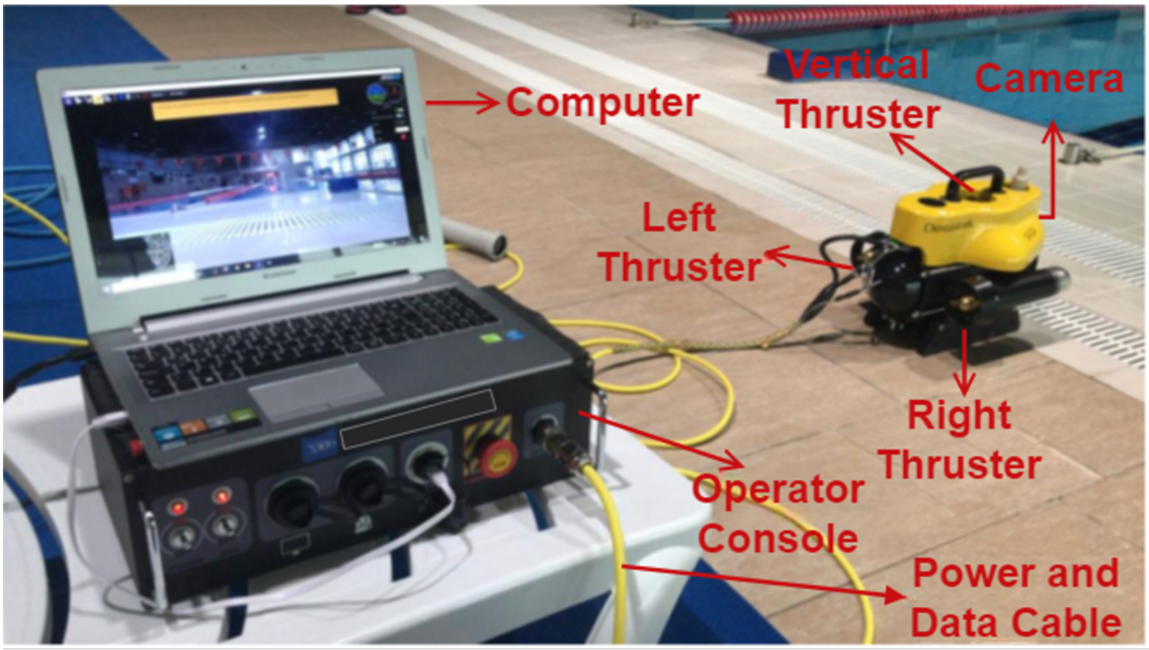

Figure 1. Unmanned underwater vehicle used in the experiment.

The experimental data related to underwater object detection was obtained from a camera integrated into the remotely operated underwater vehicle shown in Figure 1. The camera, with vertical and horizontal fields of view of 128 and 96 degrees respectively and a resolution of 700 TVL, is placed in a waterproof compartment at the front of the vehicle. This vehicle can reach depths of 200 m to perform tasks such as underwater observation, real-time high-resolution video and photo capture, data collection, and underwater mapping.

3. Underwater Pipeline Damage Detection Using Convolutional Neural Networks

In this study, damage assessment on the pipeline was conducted using pool images taken from a camera integrated into the unmanned underwater vehicle. The camera, placed in a waterproof compartment at the front of the vehicle, has a vertical field of view of 128 degrees, a horizontal field of view of 96 degrees, and a resolution of 700 TVL [51]. The damage assessment study was carried out in a pool environment based on experimental data. The video footage from the vehicle's camera was transmitted via fiber optic cable to the operator console and then to a computer via Ethernet cable. To diagnose the damage, the images were processed using a deep learning method, the Convolutional Neural Network (CNN) algorithm. The all study was conducted in a Python environment. The pipes and damages used for the pipeline damage assessment experiment are presented in Figure 2.

In our study, the type of pipe used is polyvinyl chloride pipes, which are commonly used in water transportation systems. These pipes are frequently preferred due to their strength and resistance to corrosion. In this study, polyvinyl chloride pipes were used for testing purposes. In real-world aquatic environments, biofouling, mineral deposits, wear, scratches, and UV light cause damage that manifests as color changes on the surface of underwater pipes. In marine environments, the accumulation of algae, bacteria, shellfish, and certain minerals (especially iron) results in biofouling, creating reddish-brown discolorations on the pipe surface. This buildup can damage the pipe's outer surface and negatively affect the functionality of the pipeline. Underwater PVC pipes may also experience surface scratches and abrasions due to the friction of particles carried by the water flow. These physical deformations lead to color changes and reduced performance. Additionally, chemicals present in seawater or maintenance chemicals used around the pipe can cause discoloration, impacting the material's long-term durability [60]. In this study, as seen in Figure 2, the damages identified are associated with these types of color changes observed on pipe surfaces in real-world marine environments. Specifically, damage types such as surface scratches, dents, abrasions and color changes were identified, and a Convolutional Neural Network (CNN) algorithm was trained using this visual data. The pipe deformations shown in Figures 2 and 3 focus on how our developed algorithm detects these types of damages. The damage types addressed in our study include physical deterioration that could have potential negative effects on the structural integrity and safety of the pipes.

In this study, the Faster R-CNN Inception v.2 object detection algorithm, widely used in various object detection tasks, was utilized as the object detection algorithm [61,62]. The hybrid use of the CNN model with SVM was addressed similarly to the technique mentioned in the article "A hybrid CNN–SVM classifier for weed recognition in winter rape field." In this study, the Inception v.2 part of the model was responsible for extracting feature vectors from the images. These features were then used to train the SVM classifier, yielding good results. The hardware specifications used for the training are: Windows 10 operating system, Nvidia GTX 1060 GPU, Core i5-7700 CPU, Python 3.6.5, and 16 GB RAM. With these specifications, the training process took approximately 16 h.

Figure 2. Damaged pipe used in the experiment (red-colored region is defined as “damage”).

Figure 3. Damaged pipe used in the experiment (red-colored region is defined as "damage").

3.1. Convolutional Neural Network

3.1.1. Input Layer

The size of the data provided to this layer is crucial for the success of the model. If the amount of data is too large, the training can yield very successful results but will take a long time. Conversely, if the amount of data is too small, the success rate of the training will significantly decrease. While data size in the input layer plays an important role, it is not sufficient on its own for the overall performance of the model. The number of filters, layer depth, activation functions, data variety, data preprocessing, and similar factors also play significant roles in all layers of the model and affect the model's overall performance.

3.1.2. Convolutional Layer

This layer is the first layer that extracts features from the input data. Different filters are applied to the input data. The applied filters are passed over the entire image to produce an output. The output after applying the filter is known as a feature map [63].

3.1.3. Rectified Linear Unit Layer

The Rectified Linear Unit (ReLU) is a commonly used activation function in CNN [64]. ReLU is defined as follows:

$$g(y) = max(0, y) \tag{1}$$

Here, $g(y)$ is a function corresponding to the input y. ReLU sets the negative values of the data applied to its input to zero. This reduces the computational load and training time.

3.1.4. Pooling Layer

The pooling layer is used between successive convolutional layers. The primary purpose of using this layer is to reduce the computational intensity in subsequent layers. There are various pooling methods; one commonly used method is max pooling [65]. In max pooling, the maximum values from the values corresponding to the filter are selected, and the filter moves two steps after each application area. The method used in this study is the max pooling method.

3.1.5. Fully Connected Layer

A fully connected layer connects to all nodes in all layers before and after it. The fully connected layer adds weights to the data to enable accurate classification of the data received from previous layers. After this process, the network provides predictions. Predictions are obtained by calculating probabilities between feature classes detected in previous layers. If the weighting is incorrect before producing a prediction, the predictions are incorrect and a cost function is calculated. The cost function serves as a guide to optimize our model. The cost function between the actual and predicted networks has been minimized using the backpropagation algorithm [66]. Additionally, overfitting is an undesirable condition in CNNs. The dropout method has been developed in this layer to prevent overfitting [9].

3.1.6. DropOut Layer

In CNNs, excessive training can lead to overfitting—memorization, which reduces the training error at each step during the training of the CNN model, but the test error may not decrease in the same direction. The reliability of a training with overfitting is low, and a training model that starts memorizing is formed. A training model that has overfitted will perform poorly when presented with an image outside of the training dataset. Large datasets like ImageNet have labeled data samples to prevent overfitting [9]. In this study, since the dataset created is not as large as ImageNet, dropout has been used to prevent overfitting.

3.1.7. Classifier Layer

The output value of the classification layer should be equal to the number of objects to be classified. For instance, if five classifications are to be made, the output of the layer should be five. Model predictions are assigned as values in the range of 0–1. A classification can be added to the CNN architecture or created as a separate model. There are different classifier models. In this study, CNN and SVM were used as a hybrid. CNN is a feature extractor, and SVM has been used as a classifier to recognize the object being sought [11].

SVMs choose from an infinite number of decision boundaries that minimize error with the greatest distance between two classes [67]. SVM uses the output of CNN as input and determines the classes by extracting features obtained by CNN. Thus, classification is achieved.

3.2. Convolutional Neural Network Training

To train the CNN architecture, a dataset was created using deep learning and image augmentation methods from photographs of damage taken on the pipeline. For the damage diagnosis study, the dataset from damages on the underwater pipeline was created through data augmentation without altering the characteristic features of the images. The data augmentation methods used for this study include rotating the image horizontally and vertically, rotating the image at specific angles, shifting the image horizontally and vertically, zooming, darkening, lightening, and changing the color. In CNN, each photo collected with the LabelImg program was individually labeled. The labeling process generated files with a .xml extension. After completing the labeling process, the photos were divided into two separate folders to run the training algorithm. These folders are the training and testing folders. Eighty percent of the photos of the object to be detected were placed in the training folder, and twenty percent were placed in the testing folder, and the training of the CNN was initiated. A total of 2000 data were collected, with 1500 used for training and 500 for testing purposes. Each image in the training set contains at least one damage label.

3.3. Underwater Damage Detection Performance of the CNN Algorithm

Using the CNN algorithm, damages on the underwater pipeline in a pool environment were diagnosed online using an unmanned underwater vehicle. The damage diagnosis results are presented in Figure 4. Although the damaged section on the underwater pipeline is very small, it was successfully detected using the CNN algorithm. The accuracy of detection in the pool environment can vary depending on lighting conditions, light refraction, reflections, and glare. In the pool, light usually comes from artificial sources and provides more homogeneous illumination. The reflections created by artificial lights on the water surface or direct entry of light into the water can cause glare or shadows in the image, making it difficult for the CNN algorithm to accurately recognize objects. Due to light refraction, objects may appear different from their actual form, further complicating the algorithm's ability to identify them correctly. As seen from in the Figure 4, the unmanned underwater vehicle has been endowed with the ability to detect objects through the underwater damage detection test.

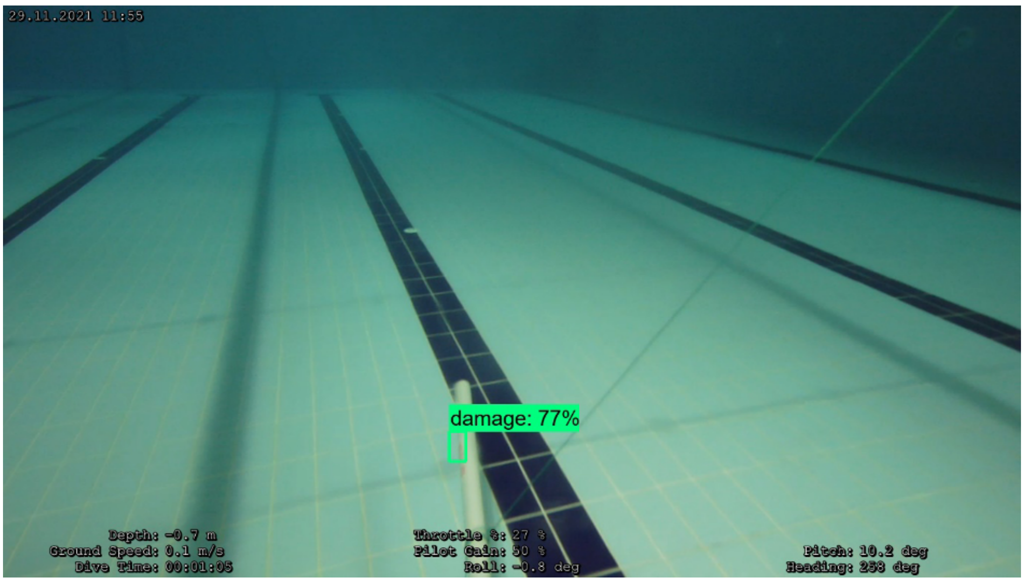

Figure 4. Pipeline damage detection experiment results.

In Figure 5, the accuracy and loss curves of the CNN training are visible. The ESA training for damage diagnosis was carried out over 50 Epochs. An epoch means that the model sees each data in the dataset once. As seen in Figure 5, both the training loss and the validation/consistency loss have been observed to decrease, which is a desired outcome.

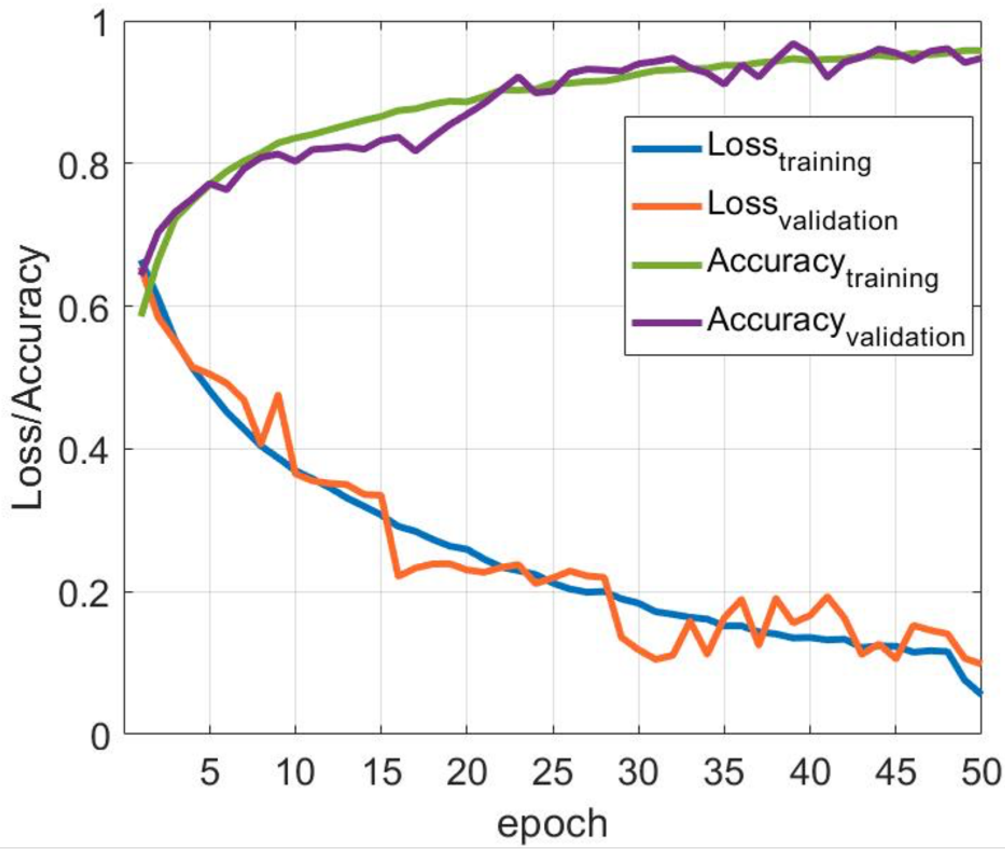

Figure 5. CNN training consistency and loss curves.

Light refraction in water, especially at different water depths, can cause image distortion. This issue can be minimized by positioning the cameras on the vehicle at appropriate angles and using multiple sensors. Additionally, adjusting the artificial light sources used in the underwater environment to the correct spectrum can also improve image quality. As depth increases, the amount of light decreases, making detection more difficult. Therefore, in deep waters, especially in deep-sea environments, more sensitive camera systems and lighting devices should be used. By using deep learning algorithms and image enhancement techniques, the quality of images obtained in low-light conditions can be improved through pre-processing. The depth and clarity of the water can cause changes in colors, making it difficult to detect real damage or anomalies in the image. To mitigate this issue, color balancing algorithms can be used. Furthermore, multi-spectral camera systems are less affected by color changes underwater, and the use of such systems can improve image quality. As in this study, deep learning-based image enhancement algorithms can make distorted underwater images clearer.

4. Autonomous Pipeline Tracking and Damage Localization with an Unmanned Underwater Vehicle

In this study, autonomous features were endowed to a remotely operated underwater vehicle, which was then used for underwater pipeline tracking in a pool environment, and damage detection on the pipe as well as the location of this damage were achieved. In the experiment, a schematic representation of the scenario created for the underwater pipeline and images from the pool experiment are given in Figures 6 and 7. As seen in Figure 6, during the experiment, an underwater pipeline was created inside the pool using one 3-m and two 2-m pipes.

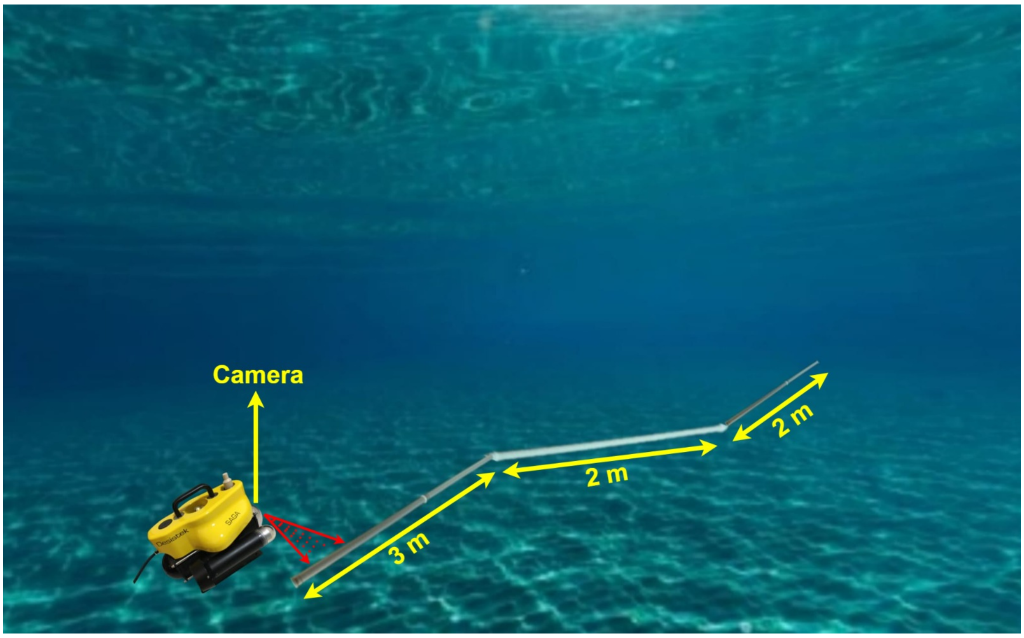

Figure 6. Pipeline tracking experiment scenario schematic representation.

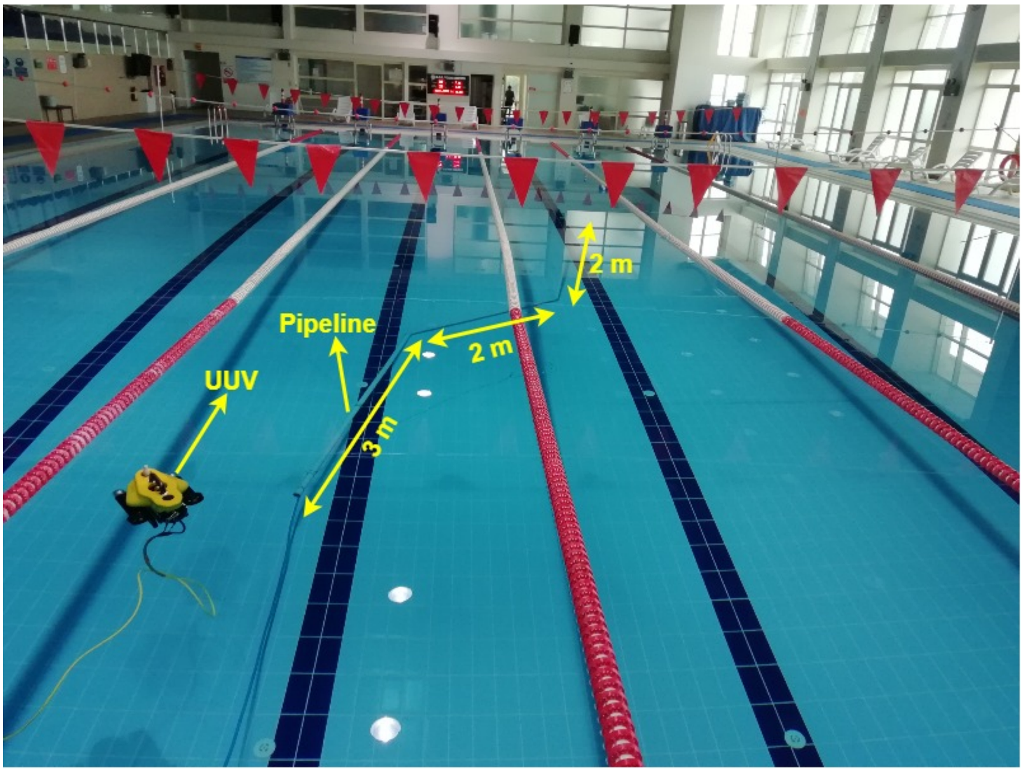

Figure 7. Pipeline tracking pool experiment.

Preliminary tests were conducted in a pool environment to enable the unmanned underwater vehicle to autonomously follow the desired pipeline. In these tests, the PWM values that need to be applied to the vehicle's thrusters were determined to enable the vehicle to autonomously perform the following movements: moving forward 3 m, turning right, moving 2 m, turning right, and then moving another 2 m in a linear and angular motions.

In these tests, the vehicle's speed information corresponding to the PWM values of the thruster motors was obtained by remotely controlling the vehicle to move along different routes. The speed information corresponding to these PWM values and, consequently, the distance information obtained were observed. The necessary PWM values for the vehicle to follow a pipeline of known length, which will be used in the experiment, were determined as a result of these tests and sent to the vehicle's thruster motors as input information. Consequently, an object tracking algorithm was developed using the data obtained from these preliminary tests to enable the vehicle to autonomously follow the desired object.

During the vehicle's tracking of the pipeline, the damages on the pipe were detected using the CNN method detailed above. The damage location detection information has been supported with the vehicle navigation and autopilot described in the next section.

4.1. Navigation of Unmanned Underwater Vehicle

The navigation information of unmanned underwater vehicle comes from IMU, depth sensor. IMU with MPU-6000 series used in the experiment combines 3-axis gyroscopes integrated with magnetic compass, 3-axis accelerometer. The depth sensor used in the experiment is the measurement specialties MS5837-30BA, which can measure up to 30 bar (300 m/1000 ft depth). In the pool experiments, the speed and orientation information of the unmanned underwater vehicle was obtained from the Inertial Measurement Unit (IMU) sensor integrated into the vehicle. The linear speed of the vehicle used in the pool experiments was obtained by integrating the acceleration. Linear position information was obtained by taking the double integral of the measurement from the accelerometer sensor. The yaw angle (rotation around the z-axis) information was obtained by taking the integral of the gyroscope measurement data once. The depth information of the vehicle from the pool surface was obtained from the pressure sensor integrated into the vehicle [68]. The equations for the vehicle's linear motion along the x and y axes and angular motion around the z-axis are given in (2)–(4) [69].

$$\dot{x} = u_r * \cos_\varphi - v_r * \sin_\varphi + \dot{x}_c \tag{2}$$

$$\dot{y} = u_r * \sin_\varphi + v_r * \cos_\varphi + \dot{y}_c \tag{3}$$

$$\dot{\varphi} = r \tag{4}$$

In this study, since the vehicle used only has forward speed, the relative velocity, v_r has been neglected, and external disturbances were disregarded in the pool experiments. Thus, the relationship between the vehicle speed and the absolute speed is reduced as in (5) and (6) [69].

$$\dot{x} = u_r * \cos_\varphi \tag{5}$$

$$\dot{y} = u_r * \sin_\varphi \tag{6}$$

4.2. Autopilot of Unmanned Underwater Vehicle

In this study, the Pixhawk control board was used for the autopilot of the unmanned underwater vehicle as seen Figure 8. Pixhawk is frequently used as a control board for unmanned aerial vehicles, surface, and underwater vehicles due to its low cost and high-performance advantages. In this study, communication between the user computer and the unmanned underwater vehicle, as well as sending inputs to the vehicle via Python software, (3.9.4 version) were carried out using the MAVLink protocol. The pymavlink library was used to establish the connection between the vehicle and Python.

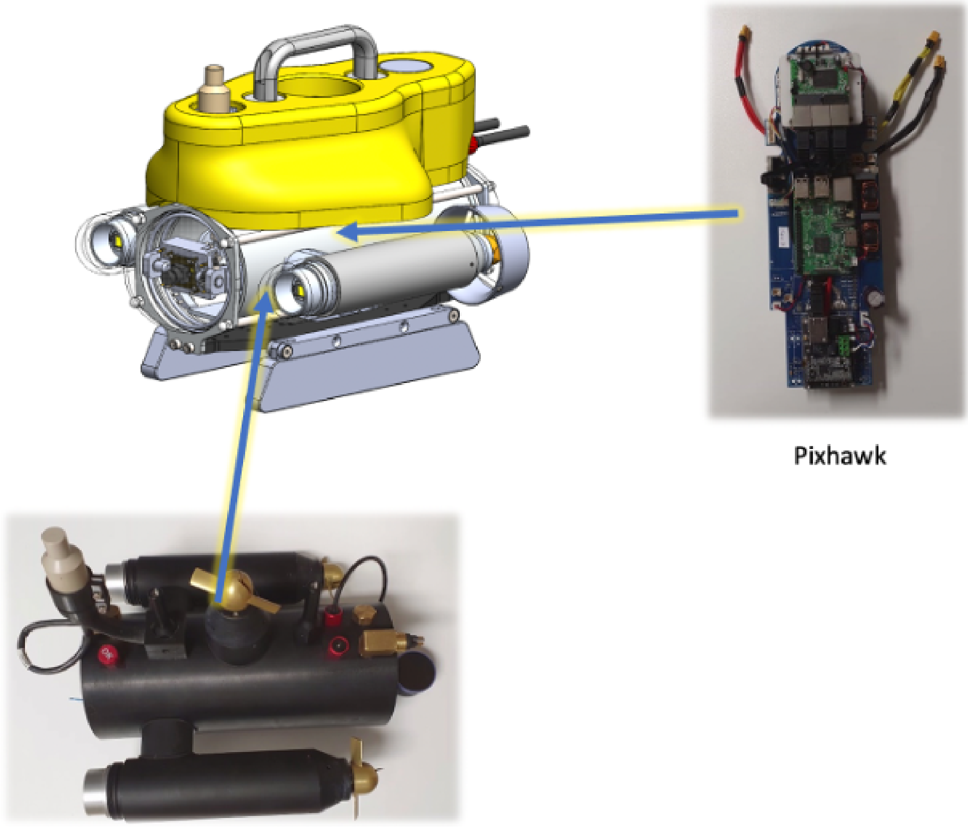

Figure 8. Vehicle's pixhawk and thrusters used in the experiment.

In the experiment, to automate the pipeline tracking, a series of preliminary pool tests were initially conducted to observe the vehicle's responses to PWM (pulse-width modulation) signals sent to the motors by the Pixhawk. Subsequently, the necessary input value of each thrusters' values to be sent to the motors for the created pipeline tracking scenarios were determined. Before the experiment for the established pipeline scenario, the Speed-PWM and Depth-PWM relationships were observed, and the necessary PWM information to enable the vehicle to follow the pipeline was sent to the vehicle.

The vehicle has successfully followed the designated route. The reference and measured forward speed information, yaw angle, and depth information while the vehicle was following the designated pipeline route are shown sequentially in Figures 9–11.

Figure 9 shows the necessary reference forward speed information for following the specified pipeline, along with the forward speed information obtained from the accelerometer during the pipeline tracking. Figure 10 presents the necessary yaw angle for the vehicle to follow the specified pipeline in the experiment, along with the yaw angle measured from the gyroscope during the experiment.

It can be seen from Figures 9–11 that the vehicle successfully followed the necessary reference forward speed, reference depth, and reference yaw angle required for autonomously tracking the specified pipeline.

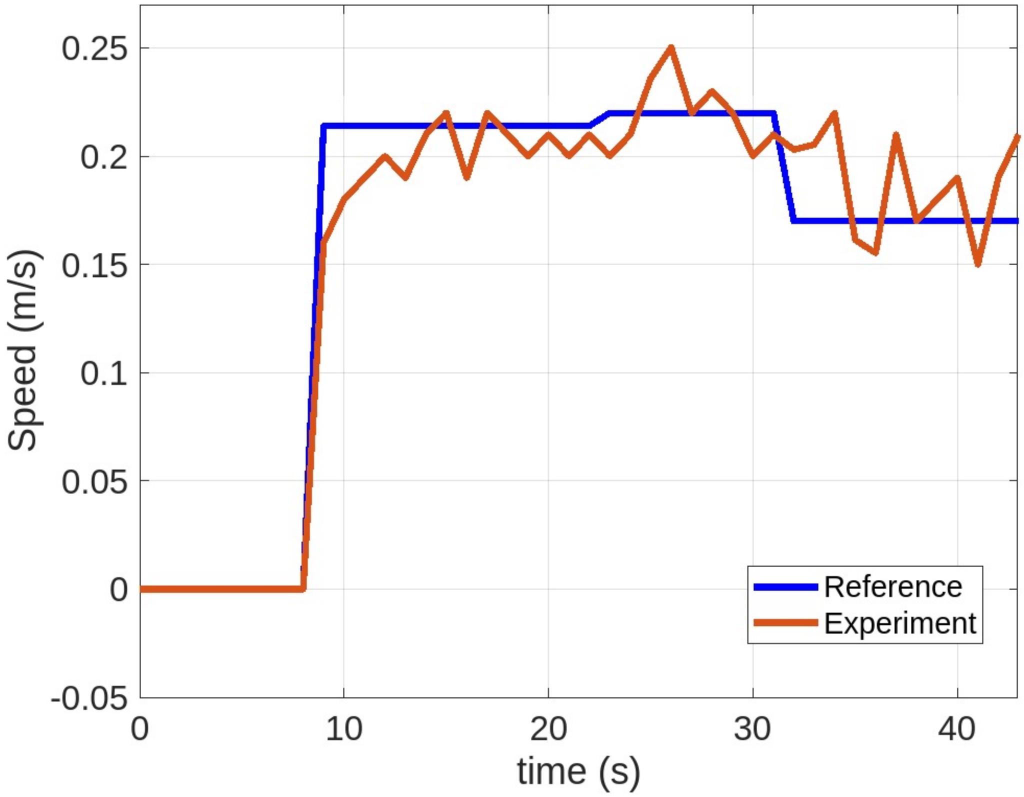

Figure 9. Reference surge speed (blue line) and measured (red line) surge speed.

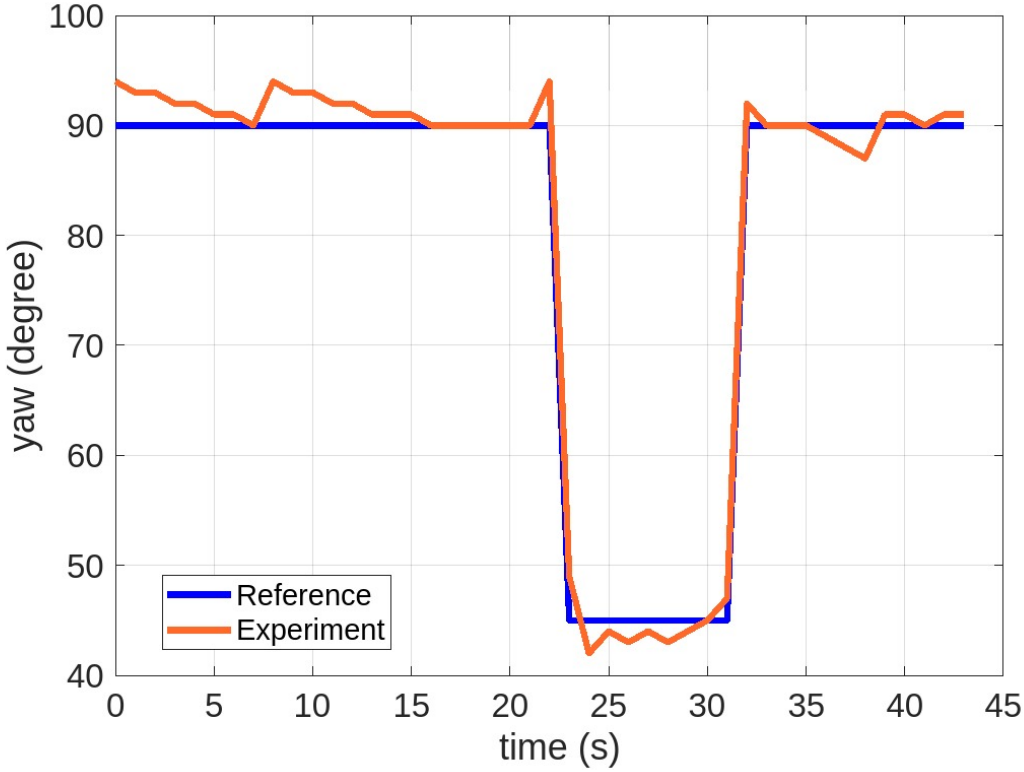

Figure 10. Reference yaw angle (blue line) and measured yaw angle (red line).

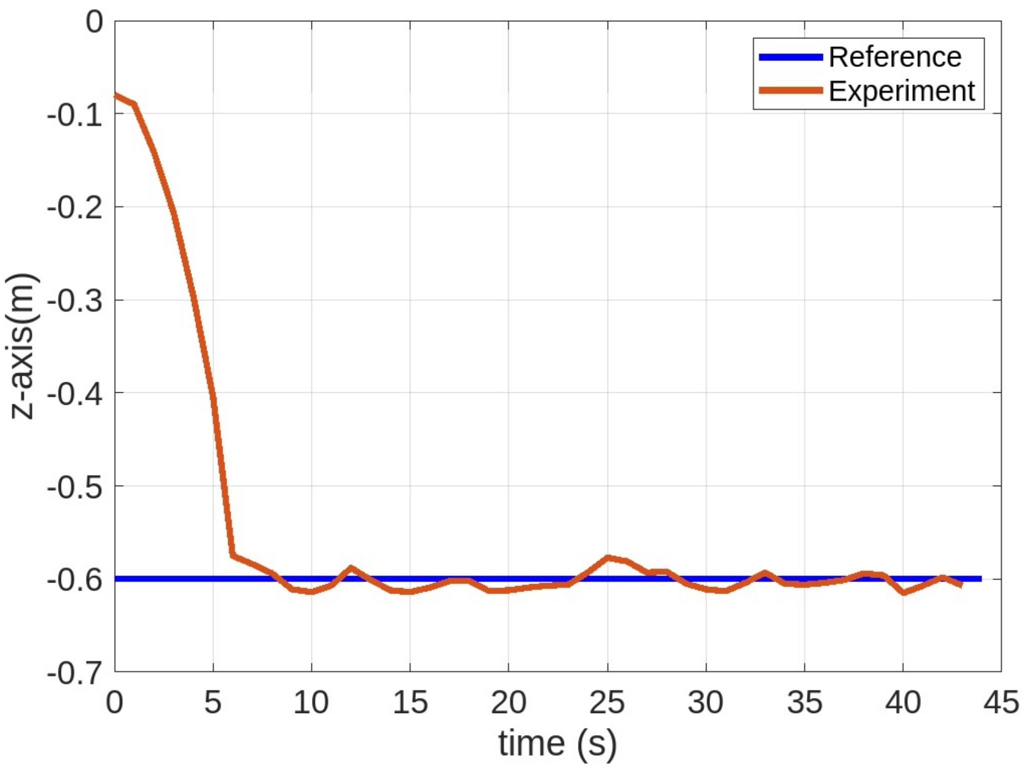

Figure 11. Reference depth value (blue line) and measured depth value (red line).

4.3. Damage Localization Using Navigation Data and Image Processing

In the pool experiment, while the unmanned underwater vehicle autonomously followed the pipeline as desired, damages on the pipeline were diagnosed using images obtained from the vehicle's camera. The position of the diagnosed damage relative to the vehicle was identified using pixel data obtained from images captured by the underwater vehicle's camera, and the real-world position was estimated based on the pixel dimensions. A dataset was created by calculating pixel values from images of damages taken from different distances. With this dataset, a support vector machine (SVM) was trained to estimate the distance corresponding to the pixel sizes, and consequently, a model was developed to predict the position of the damage. Additionally, the vehicle's position at the moment the damage was diagnosed was determined by taking the double integral of the accelerometer data, and the yaw angle was known from gyroscope measurements. By combining these two pieces of location information, the position of the damage relative to the starting point was determined.

The underwater vehicle is equipped with a specialized cable that provides both power and data transmission. This cable allows the collected data to be transmitted to a computer. The image data received by the computer is processed in the Python environment and passed through a pre-trained deep learning model. Since the pipeline tracking experiment was conducted in a pool environment, which is smaller compared to real-world scenarios, data transmission delays were neglected.

EXPERIMENT RESULTS

For this experiment, one 3-m and two 2-m pipes were placed in the pool at 45-degree angles to each other. Prior to the experiment, three separate points on the pipeline positioned in the pool were damaged. These damage images were provided in Section 1, Figure 2. The results of the damage diagnosis with CNN for the pool experiment and the location of this damage are given in Figures 12–14.

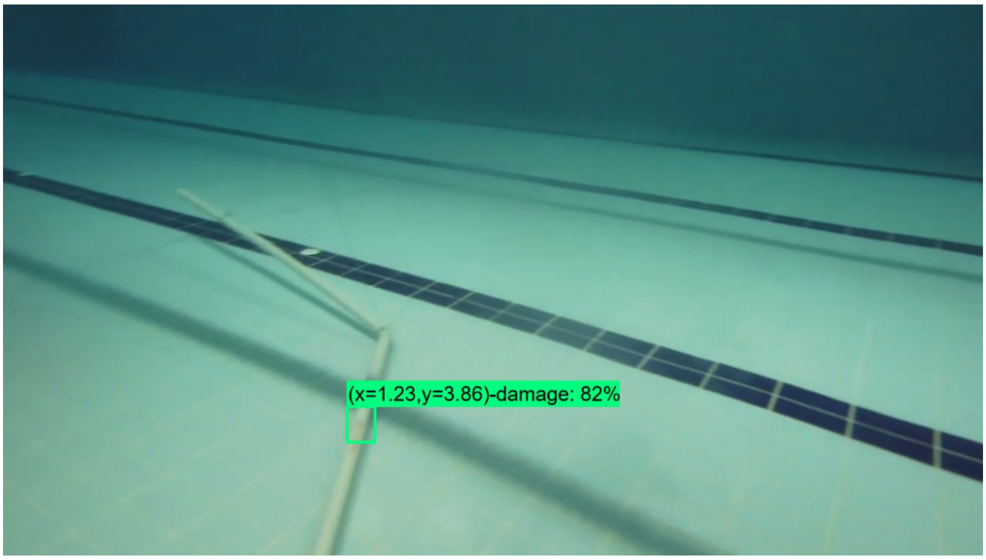

Figure 12. Location 1 of pipe line damage detection.

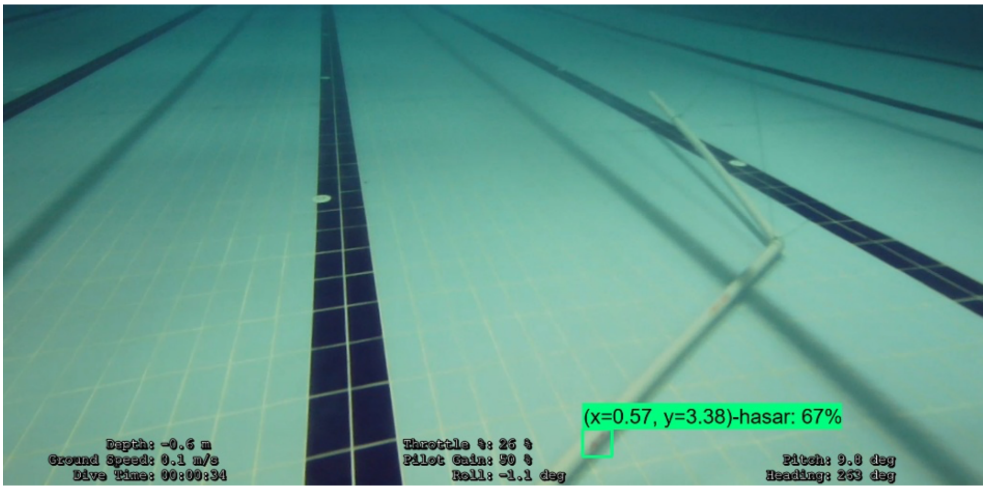

Figure 13. Location 2 of pipe line damage detection.

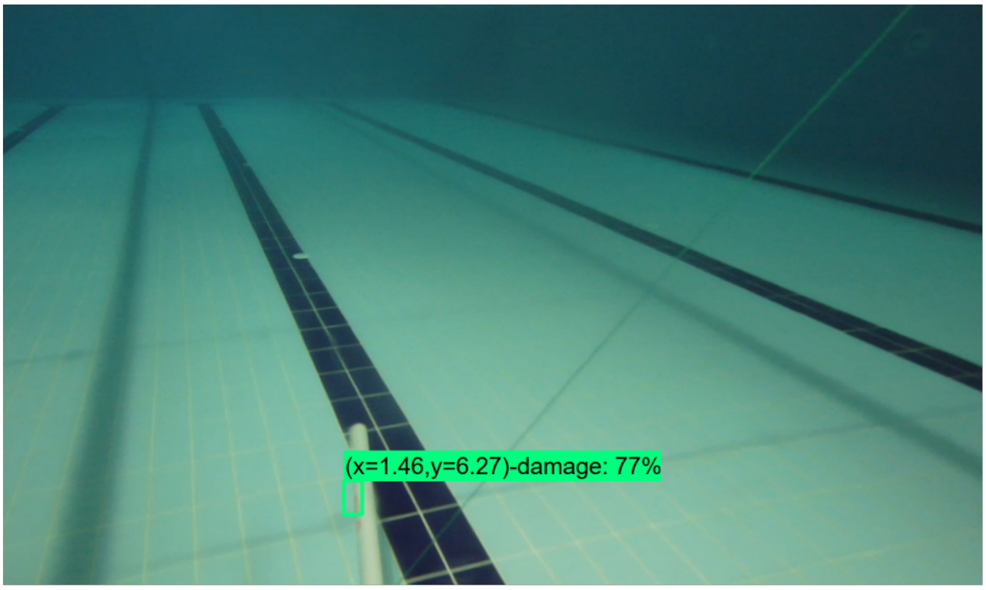

Figure 14. Location 3 of pipe line damage detection.

In Figure 15, the 3D movement of the unmanned underwater vehicle during the autonomous underwater pipeline tracking experiment are presented, respectively. The pipeline that was intended to be followed is shown as a blue line as the reference path (the known pipeline length is the actual data), and the path the vehicle followed during the experiment is shown as a red line (measured comes from navigation of the vehicle). As can be seen from Figure 15, the vehicle successfully followed the pipeline placed in the pool with high performance.

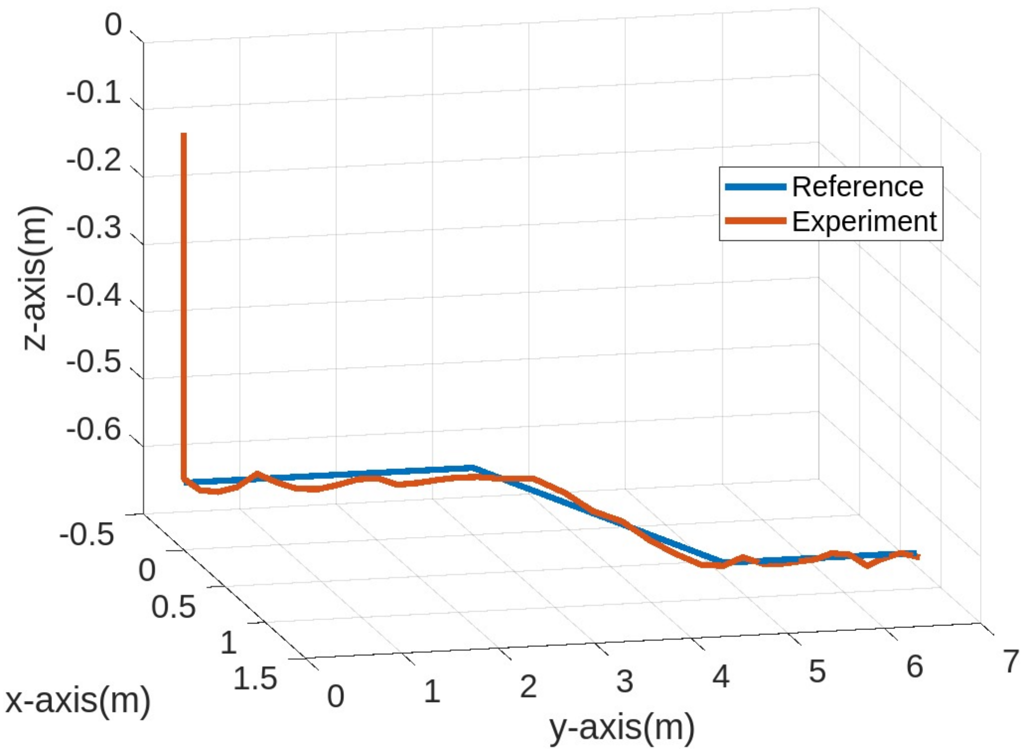

Figure 15. Reference path (blue line) and followed path (red line).

Figure 11 presents the depth information of the vehicle. In this experiment, the pipeline was placed at a depth of 1.8 m in the pool. To follow the pipeline from above at this depth, the unmanned underwater vehicle was submerged to a depth of 60 cm in the pool. Figure 10 presents the changes in the yaw angle made by the vehicle while following the pipeline. Initially, the vehicle followed the 3-m section of the pipeline at a 90-degree deviation angle in about 22 s (including 6 s of submersion time), the 2-m section at a 45-degree angle in 10 s, and the final 2-m section again at a 90-degree angle in 10 s.

Table 2 presents the root mean square error (RMSE) values for the study of underwater pipeline tracking with an autonomous unmanned underwater observation vehicle. It is the difference between the reference position value and the tracked position value. These RMSE values are known to the user as the path where the pipeline was placed, and were obtained from the path information recorded by the vehicle after completing the pipeline tracking.

Table 2. RMSE values between reference and tracked position value.

Position	RMSE
x	0.072 m
y	0.037 m
z	0.161 m
yaw	1.9 deg

5. Conclusions

The remotely controlled unmanned underwater vehicle, equipped with autonomous capabilities, successfully followed the pipeline placed in the pool. During the tracking process, damage to the pipeline was identified using a deep learning algorithm, CNN, from the images taken by the camera integrated into the vehicle. From our pool experiments, it was observed that factors such as light refraction, reflection, glare negatively affected the quality of the images obtained from the camera in underwater environments. Although the detection performance of the CNN algorithm decreased in a pool environment where light is refracted, reflected and shined environments, its object detection performance was still observed to be successful in experimentally. After detecting the damages on the autonomously tracked pipeline in the pool environment with CNN, the location of the damage was identified by correlating it with the vehicle's navigation data. The navigation data of the vehicle comes from the accelerometer, gyroscope, and pressure sensor. When comparing the actual path (reference path) values of the pipeline, which has a known length, with the measured position and deviation angle (experimental data) values during the vehicle's pipeline tracking, it was observed that the vehicle followed the pipeline with high accuracy. The results obtained in this study demonstrate that the detection and regular monitoring of defects in underwater pipelines can be carried out autonomously, safely, and continuously using unmanned underwater vehicles and deep learning algorithms.

Despite this achievement, the study has some limitations. The system in this study operates based on the images provided by the camera integrated into the unmanned underwater vehicle. Since the camera can only focus on the upper and side parts of the pipeline, it may not be possible to detect damage occurring on the underside of the pipeline. This is an important limitation, especially in cases where damage such as corrosion or cracks may occur on the lower surface of the pipeline. To overcome this limitation, multi-angle camera systems capable of imaging the lower part of the vehicle, or additional sensors capable of providing a full circular view around the pipeline, could be used. Another limitation is energy management for long-term underwater operations. Extending the battery capacity, turning off unused sensors, or processing only critical data could extend the operational duration of the vehicle. In the future work, autonomous tracking, damage detection, and damage localization of a pipeline laid in a marine environment will be tested using additional sensors such as sonar, under different scenarios and for various pipeline types.

Author Contributions: Writing, Data curation, Formal analysis, Investigation, Methodology, Visualization, S.K.K.; Software, Research, Data curation, R.F.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Scientific and Technological Research Council of Turkey (grant 119E037) and grant 2210-C.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: This work is supported by the Scientific and Technological Research Council of Turkey (grant 119E037) and 2210-C. The authors are grateful for the support of the Scientific and Technological Research Council of Turkey and project of 119E037 team members and Berna Erol Cantekin, Ş. Hakan Kutoğlu , M. Kemal Leblebicioğlu, Rıfat Hacıoğlu, K. Sedar Görmüş.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alvarez, A. Redesigning the SLOCUM Glider for Torpedo Tube Launching. *IEEE J. Ocean. Eng.* **2010**, *35*, 984–991. [CrossRef]
2. Bishop, G.C. Gravitational Field Maps and Navigational Errors. In Proceedings of the 2000 International Symposium on Underwater Technology, Tokyo, Japan, 23–26 May 2000; pp. 149–154.

3. Fattah, S.A.; Abedin, F. R3Diver: Remote Robotic Rescue Diver for Rapid Underwater Search and Rescue Operation. In Proceedings of the 2016 IEEE Region 10 Conference (TENCON), Singapore, 22–25 November 2016; pp, 3280–3283.
4. Wynn, R.B.; Huvenne, V.A.; Le Bas, T.P.; Murton, B.J.; Connelly, D.P.; Bett, B.J.; Ruhl, H.A.; Morris, K.J.; Peakall, J.; Parsons, D.R.; et al. Autonomous Underwater Vehicles (AUVs): Their Past, Present and Future Contributions to the Advancement of Marine Geoscience. *Mar. Geol.* **2014**, *352*, 451–468. [CrossRef]
5. Dinc, M.; Chingiz, H. Autonomous Underwater Vehicles. *J. Mar. Eng. Technol.* **2015**, *14*, 32–43. [CrossRef]
6. Lee, J.; Park, J.-H.; Hwang, J.-H.; Noh, K.; Choi, Y.; Suh, J. Artificial Neural Network for Glider Detection in a Marine Environment by Improving a CNN Vision Encoder. *J. Mar. Sci. Eng.* **2024**, *12*, 1106. [CrossRef]
7. Xia, T.; Cui, D.; Chu, Z.; Yu, X. Autonomous Heading Planning and Control Method of Unmanned Underwater Vehicles for Tunnel Detection. *J. Mar. Sci. Eng.* **2023**, *11*, 740. [CrossRef]
8. Kartal, S.K.; Hacıoğlu, R.; Görmüş, S.K.; Kutoğlu, Ş.H.; Leblebicioğlu, M.K. Modeling and Analysis of Sea-Surface Vehicle System for Underwater Mapping Using Single-Beam Echosounder. *J. Mar. Sci. Eng.* **2022**, *10*, 1349. [CrossRef]
9. Gu, J.; Wang, Z. Recent Advances in Convolutional Neural Networks. *Pattern Recognit.* **2018**, *77*, 354–377. [CrossRef]
10. Kaya Ustabaş, G.; Kocabaş, S.; Kartal, S.; Kaya, H.; Tekin, I.Ö.; Tığlı Aydin, R.S.; Kutoğlu, Ş.H. Detection of airborne nanoparticles with lateral shearing digital holographic microscopy. *Opt. Lasers Eng.* **2022**, *151*, 106934. . [CrossRef]
11. Niu, X.; Ching, Y.S. A Novel Hybrid CNN-SVM Classifier For Recognizing Handwritten Digits | Pattern Recognition. *Pattern Recognit.* **2011**, *45*, 131825.
12. Russakovsky, O.; Deng, J. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
13. Simonyan, K.; Andrew, Z. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
14. Spahic, R.; Poolla, K. Image-based and Risk-informed Detection of Subsea Pipeline Damage. *Discov. Artif. Intell.* **2023**, *3*, 23. [CrossRef]
15. Bastian, B.; Jaspreeth, N. Visual Inspection and Characterization of External Corrosion in Pipelines Using Deep Neural Network. *NDT E Int.* **2019**, *107*, 102134. [CrossRef]
16. Avci, A.; Karadeniz Kartal, S. Leakage Detection in Underwater Oil and Natural Gas Pipelines Using Convolutional Neural Networks *Int. J. Energy Appl. Technol.* **2021**, *8*, 197–202.
17. Nadimi, N.; Javidan, R. Efficient Detection of Underwater Natural Gas Pipeline Leak Based on Synthetic Aperture Sonar (SAS) Systems. *Mar. Sci. Eng.* **2021**, *9*, 1273. [CrossRef]
18. Shi, Y.; Zhang, C. Theory and Application of Magnetic Flux Leakage Pipeline Detection. *Sensors* **2015**, *15*, 31036–31055. [CrossRef]
19. Hong, X.; Huang, L. Shedding Damage Detection of Metal Underwater Pipeline External Anticorrosive Coating by Ultrasonic Imaging Based on HOG + SVM. *Mar. Sci. Eng.* **2021**, *9*, 364. [CrossRef]
20. Bertulessi, M.; Bignami, D. Experimental Investigations of Distributed Fiber Optic Sensors for Water Pipeline Monitoring. *Sensors* **2023**, *23*, 6205. [CrossRef]
21. Peng, X.; Hao, H. A Numerıcal Study Of Damage Detectıon Of Underwater Pıpelıne Usıng Vıbratıon-Based Method. *Int. J. Struct. Stab. Dyn.* **2012**, *12*, 1250021. [CrossRef]
22. Shitong, Z.; Yanhui, W. Subsea Pipeline Leak Inspection by Autonomous Underwater Vehicle. *Appl. Ocean. Res.* **2020**, *107*, 102321.
23. Khan, A.; Azhar, S. Visual Feedback–based Heading Control of Autonomous Underwater Vehicle for Pipeline Corrosion Inspection. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 1729881416658171. [CrossRef]
24. Foresti, G.L. Visual Inspection of Sea Bottom Structures by an Autonomous Underwater Vehicle. *Int. J. Adv. Robot. Syst.* **2001**, *31*, 691–705. [CrossRef] [PubMed]
25. Cang, X.; Jouvencel, B. Coordinated Formation Control of Multiple Autonomous Underwater Vehicle for Pipeline Inspection. *Int. J. Adv. Robot. Syst.* **2010**, *7*, 3.
26. Cang, X.; Jouvencel, B. An Autonomous Underwater Vehicle Simulation With Fuzzy Sensor Fusion for Pipeline Inspection. *IEEE Sensors J.* **2023**, *23*, 8941–8951.
27. Szegedy, C.; Liu, W. Going Deeper With Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp.1–9.
28. He, X.; Wei, Z. Emotion Recognition By Assisted Learning With Convolutional Neural Networks. *Neurocomputing* **2018**, *291*, 187–194. [CrossRef]
29. Pertusa, A.; Gallego, A. MirBot: A Collaborative Object Recognition System for Smartphones Using Convolutional Neural Networks. *Neurocomputing* **2018**, *293*, 87–99. [CrossRef]
30. Li, X.; Min, S. Fast Accurate Fish Detection and Recognition Of Underwater İmages with Fast R-CNN. In *OCEANS MTS/IEEE Washington*; IEEE: Piscataway, NJ, USA, 2015.
31. Chavez, A.G.; Birk, A. Stereo-Vision Based Diver Pose Estimation Using LSTM Recurrent Neural Networks For AUV Navigation Guidance. In *OCEANS-Aberdeen*; IEEE: Piscataway, NJ, USA, 2017; pp.1–7.
32. Buß, M.; Steiniger, Y. Hand-Crafted Feature Based Classification against Convolutional Neural Networks for False Alarm Reduction on Active Diver Detection Sonar Data. In *OCEANS MTS/IEEE Charleston*; IEEE: Piscataway, NJ, USA, 2018; pp.1–7.
33. Williams, D.P. *Demystifying Deep Convolutional Neural Networks For Sonar Image Classification*; NATO STO Centre for Maritime Research and Experimentation (CMRE) Viale San Bartolomeo: La Spezia, Italy, 2019; Volume 400, pp. 513–520.
34. Williams, D.P. On the Use of Tiny Convolutional Neural Networks for Human-Expert-Level Classification Performance in Sonar Imagery. *IEEE J. Ocean. Eng.* **2020**, *4*, 236–260. [CrossRef]

35. Cortes, C.; Vapnik, V. *Support-Vector Networks Machine Learning*; Springer: Cham, Switzerland, 1995; Volume 20, pp. 273–297.
36. Zhang Z., Ding S. MBSVR: Multiple birth support vector regressions. *Inf. Sci.* **2021**, *552*, 65–79. [CrossRef]
37. Zhao, Q.; Qin, X. A novel prediction method based on the support vector regression for the remaining useful life of lithium-ion batteries. *Microelectron. Reliab.* **2018**, *85*, 99–108. [CrossRef]
38. Li, X.; Shu, X. An On-Board Remaining Useful Life Estimation Algorithm for Lithium-Ion Batteries of Electric Vehicles. *Energies* **2017**, *10*, 691. [CrossRef]
39. Meniconi, S.; Brunone, B.; Tirello, L.; Rubin, A.; Cifrodelli, M.; Capponi, C. Transient tests for checking the Trieste subsea pipeline: Towards the field tests. *J. Mar. Sci. Eng.* **2024**, *12*, 374. [CrossRef]
40. Meniconi, S.; Brunone, B.; Tirello, L.; Rubin, A.; Cifrodelli, M.; Capponi, C. Transient tests for checking the Trieste subsea pipeline: Diving into fault detection. *J. Mar. Sci. Eng.* **2024**, *12*, 391. [CrossRef]
41. Oktanisa, I.; Mahmudy, W.F. Inflation Rate Prediction in Indonesia using Optimized Support Vector Regression Model. *J. Inf. Technol. Comput. Sci.* **2020**, *5*, 104–114. [CrossRef]
42. Manasa, J.; Grupta, R. Machine Learning based Predicting House Prices using Regression Techniques. In Proceedings of the 2nd International Conference on Innovative Mechanisms for Industry Applications, (ICIMIA), Bangalore, India, 5–7 March 2020; pp. 624–630.
43. Smola, A.J.; Schölkopf, B. A Tutorial On Support Vector Regression. *Stat. Comput.* **2004**, *14*, 199–222. [CrossRef]
44. Li, M.W.; Geng, J. Periodogram Estimation Based On LSSVR-CCPSO Compensation For Forecasting Ship Motion. *Nonlinear Dyn.* **2019**, *97*, 2579–2594. [CrossRef]
45. Dong, Y.; Zhang, Z. A Hybrid Seasonal Mechanism With A Chaotic Cuckoo Search Algorithm with A Support Vector Regression Model For Electric Load Forecasting. *Energies* **2018**, *11*, 1–21. [CrossRef]
46. Cheng, K.; Lu, Z. Active Learning Bayesian Support Vector Regression Model For Global Approximation. *Inf. Sci.* **2021**, *544*, 549–563. [CrossRef]
47. Zhang, Z.; Ding, S. A Support Vector Regression Model Hybridized With Chaotic Krill Herd Algorithm And Empirical Mode Decomposition For Regression Task. *Neurocomputing* **2020**, *410*, 185–201. [CrossRef]
48. Erol, B.; Cantekin, R.; Kartal, S.K.; Hacioglu, R.; Gormus, S.; Kutoglu, H.; Leblebicioglu, K. Estimation of Unmanned Underwater Vehicle Motion with Kalman Filter and Improvement by Machine Learning. *Int. J. Adv. Eng. Pure Sci.* **2021**, *33*, 67–77.
49. Liang, Z.; Wang, K. Multisensor Fusion Simultaneous Localization and Mapping System Based on Image Enhancement. *J. Mar. Sci. Eng.* **2024**, *17*, 1170. [CrossRef]
50. Wang, C.; Cheng, C.; Cao, C.; Guo, X.; Pan, G.; Zhang, F. An Invariant Filtering Method Based on Frame Transformed for Underwater INS/DVL/PS Navigation. *J. Mar. Sci. Eng.* **2024**, *12*, 1178. [CrossRef]
51. Kartal, S.; Leblebicioğlu, M.K. Experimental Test of the Acoustic-Based Navigation and System Detection of an Unmanned Underwater Survey Vehicle (SAGA). *Trans. Inst. Meas. Control* **2018**, *40*, 247–687.
52. Li, M.; Ji, H. Underwater Object Detection And Tracking Based On Multi-Beam Sonar İmage Processing. In Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 19–13 December 2013; pp. 1071–1076.
53. Mandic, F.; Ivor, R. Underwater Object Tracking Using Sonar and USBL Measurements. *J. Sensors* **2016**, *2016*, 8070286. [CrossRef]
54. Xiang, X.; Caoyang, Y. Autonomous Underwater Vehicle with Magnetic Sensing Guidance. *Sensors* **2016**, *16*, 1335. [CrossRef]
55. Bigoni, C.; Hesthaven, J. Simulation-Based Anomaly Detection and Damage Localization: An Application to Structural Health Monitoring. *Comput. Methods Appl. Mech. Eng.* **2020**, *363*, 112896. [CrossRef]
56. Katija, K.; Roberts, P. Visual Tracking Of Deepwater Animals Using Machine Learning-Controlled Robotic Underwater Vehicles. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 3–7 January 2021; pp. 859–868. [CrossRef]
57. Packard, G.E.; Kukulya, A. Continuous Autonomous Tracking And İmaging Of White Sharks And Basking Sharks Using a REMUS-100 AUV. In *OCEANS*; IEEE: Piscataway, NJ, USA, 2013; pp.1–5.
58. Kezebou, L.; Oludare, V. Underwater Object Tracking Benchmark and Dataset. In Proceedings of the IEEE International Symposium on Technologies for Homeland Security (HST), Woburn, MA, USA, 23–24 October 2019; pp.1–6.
59. Kartal, S.; Leblebicioğlu, M.K. Experimental test of vision-based navigation and system identification of an unmanned underwater survey vehicle (SAGA) for the yaw motion. *Trans. Inst. Meas. Control* **2019**, *41*, 2160–2170. [CrossRef]
60. Khan, A.; Ali, S.S.A.; Anwer, A.; Adil, S.H.; Mériaudeau, F. Subsea Pipeline Corrosion Estimation by Restoring and Enhancing Degraded Underwater Images. *IEEE Access* **2018**, *6*, 40585–40601. [CrossRef]
61. Hong, S.-J.; Kim, S.-Y. Moth Detection from Pheromone Trap Images Using Deep Learning Object Detectors. *Agriculture* **2020**, *10*, 170. [CrossRef]
62. Ning, W.; Yuanyuan, W. Marine vessel detection dataset and benchmark for unmanned surface vehicles. *Appl. Ocean. Res.* **2024**, *142*, 103835.
63. He, K.; Xiangyu, Z. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 346–361.
64. Nair, V.; Geoffrey, E.H. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the International Conference on Machine Learning, ICML10, Omnipress, Haifa, Israel, 21–24 June 2010; Volume 27, pp. 807–814.
65. Albawi, S.; Tareq, A.M. Understanding of a Convolutional Neural Network. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 16–21.

66. Albawi, S.; Tareq, A.M. Artificial Convolution Neural Network for Medical Image Pattern Recognition. *Neural Netw.* **1995**, *8*, 1201–1214.
67. De Oliveira, D.C.; Wehrmeister, M.A. Using Deep Learning and Low-Cost RGB and Thermal Cameras to Detect Pedestrians in Aerial Images Captured by Multirotor UAV. *Sensors* **2018**, *18*, 2244. [CrossRef] [PubMed]
68. Caccia, M.; Bibuli, R. Basic Navigation, Guidance And Control Of An Unmanned Surface Vehicle. *Auton. Robot.* **2008**, *25*, 349–365. [CrossRef]
69. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; Wiley: Hoboken, NJ, USA, 1999.